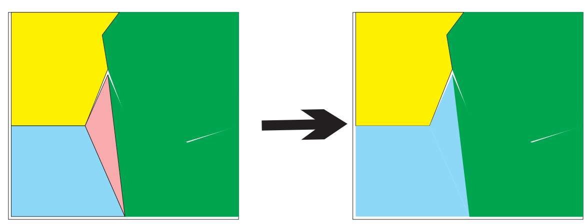
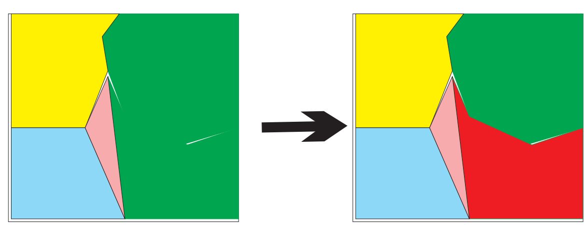
Region Segmentation

- Connected components analysis often results in many small disjointed regions.
- A connection (or break) at a single pixel can split (or merge) entire regions.
- There are three basic approaches to segmentation:
  - Region Merging - recursively merge regions that are similar.



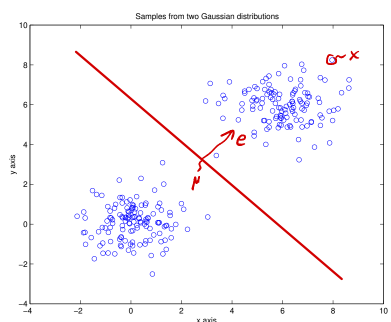
  - Region Splitting - recursively divide regions that are heterogeneous.



  - Split and merge - iteratively split and merge regions to form the "best" segmentation.

Hierarchical Clustering

- Clustering refers to techniques for separating data samples into sets with distinct characteristics.



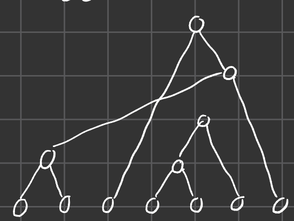
$(x-\mu)e \lessgtr 0$

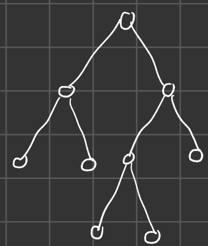
- Clustering methods are analogous to segmentation methods.
  - Agglomerative clustering - "bottom up" procedure for recursively merging clusters ⇒ analogous to region merging
  - Divisive clustering - "top down" procedure for recursively splitting clusters ⇒ analogous to region splitting

How to determine size of autocovariance matrix ??

Agglomerative ↑

Divisive ↓



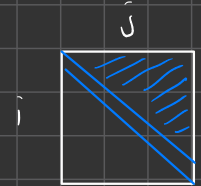


Both methods result in a tree

Depending on how you "prune", you get a different set of clusters

- could implement a threshold or stopping criteria
- divisive usually fast

N points: $\binom{N}{2}$ unique pairs $\binom{N}{2} = \frac{N(1-N)}{2}$

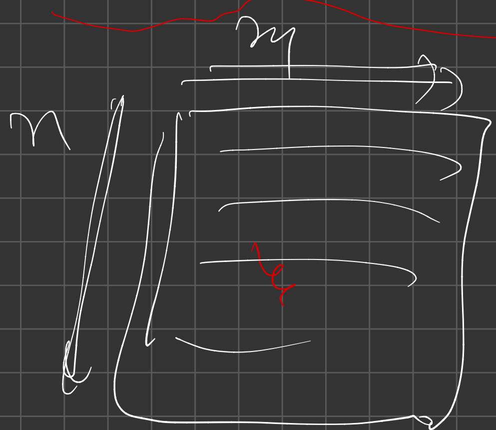


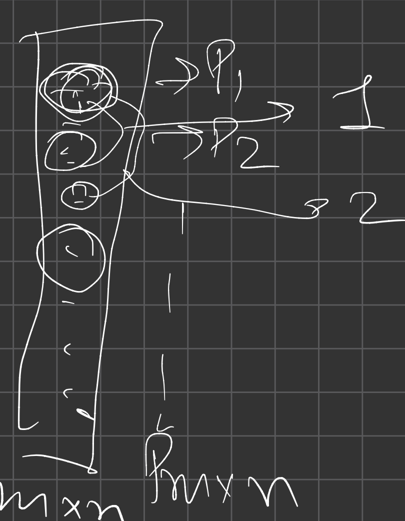
$\frac{N^2 - N}{2}$

order of complexity → (May Be Exam Question)

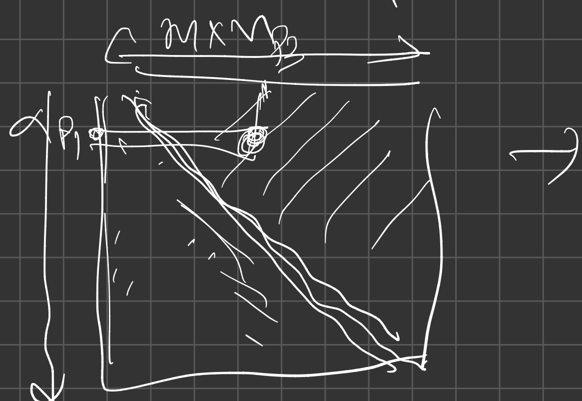
- agglomerative order of N^2 - tends to be slower
- divisive poor quality
- agglomerative high quality

$\frac{N(\cancel{N-1})}{2(\cancel{N-1})} = \frac{N}{2}$





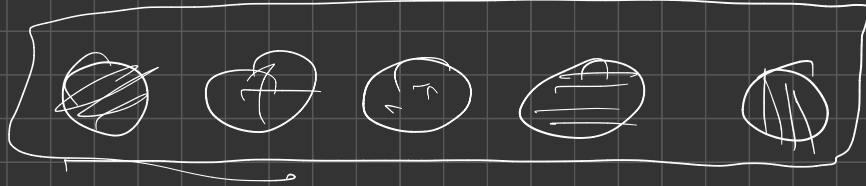
$N = m \times n$



$^{n}C_2 \to \binom{n}{2}$

$^{N}C_2 = \frac{N!}{2(N-2)!}$

$= \frac{\cancel{(N-2)!}\,(N-1)N}{2\cancel{(N-2)!}}$



$^{5}C_2 =$

★ □ ○

★ □

□ ○

★ ○

$^{3}C_2 = 3$

$$^{N}C_2 = \frac{N!}{2!\,(N-2)!} = \frac{\cancel{(N-2)!}\,(N-1)N}{2!\,\cancel{(N-2)!}} = \frac{N(N-1)}{2}$$

$3! = 1\times2\times3$

$$5! = 1\times2\times3\times4\times5$$
$$= 2!\times3\times4\times5$$
$$= 4!\times5$$
$$= 2!\times4\times5$$

Image Regions and Partitions

- Let $R_m \subset S$ denote a region of the image where $m \in \mathcal{M}$.
- We say that $\{R_m | m \in \mathcal{M}\}$ **partitions** the image if

$$\begin{aligned} \text{For all } \ m \neq k, \quad R_m \cap R_k \ &= \ \emptyset \\ \bigcup_{m \in \mathcal{M}} R_m \ &= \ S \end{aligned}$$

- Each region R_m has **features** that characterize it.

Typical Region Features

- Color
  - Mean RGB value
  - 1-D color histograms in R, G, and B
  - 3-D color histogram in (R,G,B)
- Texture
  - Spatial autocorrelation
  - Joint probability distribution for neighboring pixels (e.g. the spatial co-occurrence matrix)
  - Wavelet transform coefficients
- Shape
  - Number of pixels
  - Width and height attributes
  - Boundary smoothness attributes
  - Adjacent region labels

Recursive Feature Computation

- Any two regions may be merged into a new region.

$$R_{new} = R_k \cup R_l$$

- Let $f_n = f(R_n) \in \mathbb{R}^k$ be a k dimensional feature vector extracted from the region R_n.
- Ideally, the features of merged regions may be computed without reference to the original pixels in the region.

$$\begin{aligned} f(R_k \cup R_l) &= f(R_k) \oplus f(R_l) \\ f_{new} &= f_k \oplus f_l \end{aligned}$$

here $\oplus$ denotes some operation on the values of the two feature vectors.

Example of Recursive Feature Computation

Example: Let $f(R_k) = (N_k, \mu_k, c_k)$ where

$$
\begin{aligned}
\text{\# pixels: } N_k &= |R_k| \\
\text{mean: } \mu_k &= \frac{1}{N_k} \sum_{s \in R_k} x_s \\
\text{centroid: } c_k &= \frac{1}{N_k} \sum_{s \in R_k} s
\end{aligned}
$$

We may compute the region features for $R_{new} = R_k \cup R_l$ using the recursions

$$
\begin{aligned}
N_{new} &= N_k + N_l \\
\mu_{new} &= \frac{N_k \mu_k + N_l \mu_l}{N_{new}} \\
c_{new} &= \frac{N_k c_k + N_l c_l}{N_{new}}
\end{aligned}
$$

Review: feature vectors

k-means method/algorithm

Recursive Merging

- Define a distance function between regions. In general, this function has the form

$$d_{k,l} = D(R_k, R_l) \geq 0$$

- Ideally, $D(R_k, R_l)$ is **only** a function of the feature vectors f_k and f_l.

$$d_{k,l} = D(f_k, f_l) \geq 0$$

- Then merge regions with minimum distance.

Example of Merging Criteria

- Distance between color means

$$d_{k,l} = \frac{N_k}{N_{new}}|\mu_k - \mu_{new}|^2 + \frac{N_l}{N_{new}}|\mu_l - \mu_{new}|^2$$

- Distance between region centers

$$d_{k,l} = \frac{N_k}{N_{new}}|c_k - c_{new}|^2 + \frac{N_l}{N_{new}}|c_l - c_{new}|^2$$

- Distance formed by a weighted combination of the two

$$\begin{aligned} d_{k,l} \ &= \ \alpha \left(\frac{N_k}{N_{new}}|\mu_k - \mu_{new}|^2 + \frac{N_l}{N_{new}}|\mu_l - \mu_{new}|^2 \right) \\ &+ \ \beta \left(\frac{N_k}{N_{new}}|c_k - c_{new}|^2 + \frac{N_l}{N_{new}}|c_l - c_{new}|^2 \right) \end{aligned}$$

Recursive Merging Algorithm

- Define a distance function between regions

$$d_{k,l} = D(f(R_k), f(R_l)) > 0$$

Repeat until $|\mathcal{M}| = 1$ {

Determine the minimum distance regions

$$(k^*, l^*) = \arg \min_{k,l \in \mathcal{M}} \{d_{k,l}\}$$

Merge the minimum distance regions

$$R_{k^*} \leftarrow R_{k^*} \cup R_{l^*}$$

Remove unused region

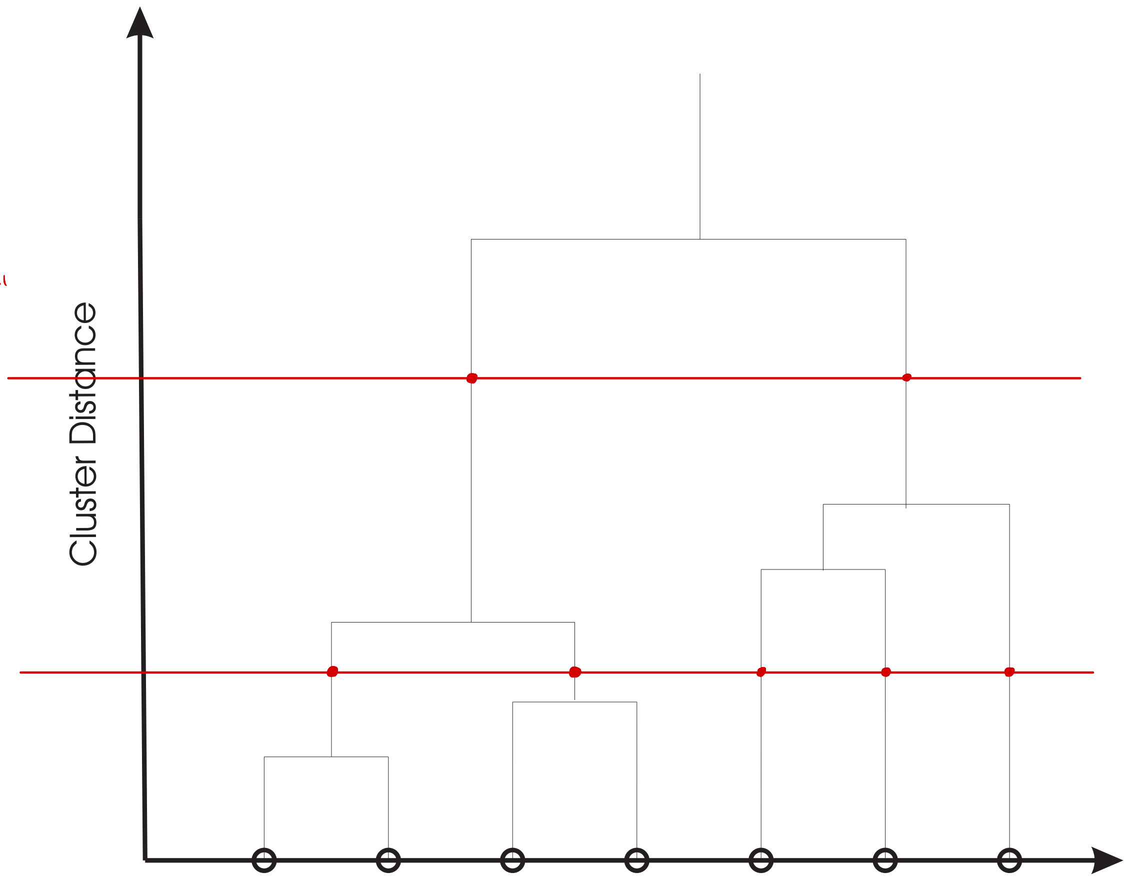
$$\mathcal{M} \leftarrow \mathcal{M} - \{l^*\}$$

}

- This recursion generates a binary tree.

Merging Hierarchy and Order Identification

set "height"
or
threshold



- Clustering can be terminated when the distance exceeds a threshold

$$d_{k^*,l^*} > Threshold \Rightarrow \text{Stop clustering}$$

- Different thresholds result in different numbers of clusters.